Robust Visual Fiducials for Skin-to-Skin Relative Ship Pose Estimation

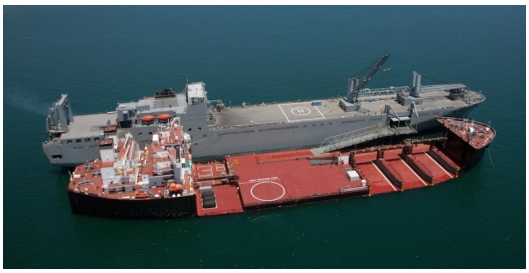
Joshua G. Mangelson, Ryan W. Wolcott, Paul Ozog, and Ryan M. Eustice

***Abstract*— This paper reports on an optical visual fiducial system developed for relative-pose estimation of two ships at sea. Visual fiducials are ubiquitous in the robotics literature, however none are specifically designed for use in outdoor lighting conditions. Blooming of the CCD causes a significant bias in the estimated pose of square tags that use the outer corners as point correspondences. In this paper, we augment existing state-of-the-art visual fiducials with a border of circles that enables high accuracy, robust pose estimation. We also present a methodology for characterizing tag measurement uncertainty on a per measurement basis. We integrate these methods into a relative ship motion estimation system and support our results using outdoor imagery and field data collected aboard the USNS John Glenn and USNS Bob Hope during skin-to-skin operations.**

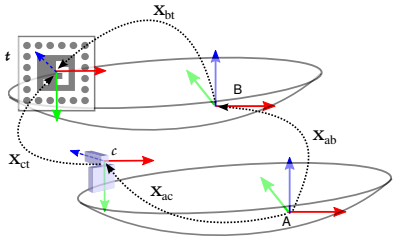
I. Introduction

Skin-to-skin ship operations are becoming more prevalent. By mooring ships hull-to-hull, cargo such as vehicles, personnel, and supplies can be quickly transferred from one ship to another without the need of a port. This process promises to significantly decrease cost and increase efficiency in both the commercial and military shipping domains (see Fig. 1(a)). However, to safely facilitate the mooring and transfer process, it is essential to have an accurate real-time estimate of the relative-pose (position and orientation) of the two ships. Optical cameras and visual fiducials can be used to directly estimate relative-pose in real time.

Over the past twenty years, roboticists and augmented reality researchers have introduced dozens of visual fiducial frameworks, though few of them are specifically designed to handle dynamic outdoor lighting conditions. Almost all of these methods seek to estimate a set of points in the 2D image frame that correspond to known 3D tag coordinates. Once found, these points can be used to estimate the pose of the tag relative to the camera. The dynamic lighting in outdoor environments often induces blooming of the camera charge-coupled device (CCD) light sensor and biases the estimation of the 2D image points, which in turn causes a significant bias in the estimated pose. This bias is especially pronounced in tag frameworks that attempt to detect corners of square tags. In addition, incorporating the relative-pose measurements derived from these tags into a filtering framework requires an accurate estimate of measurement uncertainty, which has not been heavily treated in prior visual fiducial literature.



(a) Skin-to-Skin Ship Operations



(b) Coordinate Transforms in Our System

Fig. 1. (a) Depiction of the USNS Bob Hope and a mobile landing platform ship moored skin-to-skin during testing off the coast of Long Beach, California. (b) Coordinate transforms of our suggested relative ship motion system. A camera placed on one ship measures the relative pose of a tag mounted on the other ship. If the pose of the tag and camera relative to their own respective ship is known, the relative pose of the two ships can be estimated.

In this work, we augment existing state-of-the-art visual fiducials with a border of circles that can then be used for blooming-robust, high-accuracy pose estimation. We overcome the blooming bias by using circle centers to form our 2D-3D point correspondences rather than the outer corners of squares. We are also able to increase pose estimation accuracy by increasing the number of points used in the estimation process. Fig. 1(b) shows a cartoon example of the coordinate transforms involved in our relative ship pose estimation system.

The main contributions of our work include:

- A new fiducial design that uses circle centers to enable pose estimation that is robust to dynamic lighting changes.
- A methodology for characterizing tag pose measurement uncertainty on a per measurement basis.
- Experiments using real outdoor imagery and preliminary skin-to-skin field trial results of the relative ship motion system we developed.

*This work was supported by the Office of Naval Research under award N00014-11-D-0370;

J. Mangelson is with the Robotics Institute at the University of Michigan, Ann Arbor, MI 48109, USA mangelso@umich.edu.

Ryan W. Wolcott and Paul Ozog are affiliated with the Department of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor, MI 48109, USA rwolcott@umich.edu, paulozog@umich.edu.

R. Eustice is with the Department of Naval Architecture and Marine Engineering at the University of Michigan, Ann Arbor, MI 48109, USA eustice@umich.edu.

II. Prior Work

Over the last few decades, several tags have been introduced offering a wide variety of patterns and detection strategies. In 1999, Kato and Billinghurst [1] developed the ARToolkit system that detected square visual fiducial tags and then superimposed computer graphics over the top. In 2005, Fiala [2] developed ARtags, which extended ARToolkit by using coding theory and a modified edge detection algorithm to enable robust, unique identification of individual tags and increase robustness to occlusion. In 2011, Olson [3] released a fully open-source tag suite called AprilTags, which claims to outperform ARTags in both pose detection accuracy and robustness to false positives and rotation. Each of these tag designs consist of a black square tag placed on a white background and use the four outer tag corners to estimate pose, and are thus susceptable to CCD blooming bias. Our proposed method overcomes this bias through the use of circle centers.

Other researchers have proposed the use of circular designs. In 2004, Chen et al. [4] proposed a method to determine the normal and center (up-to a sign parameter) of a circle detected in the image plane as an ellipse from a single view by fitting a conic to the detected ellipse. In 2006, Rice et al. [5] presented Cantag, another open-source tag software suite with both circular and square-shaped tags along with a variety of detection algorithms. In 2011, Pagani et al. [6] proposed a tag design with a single outer ring and two inner rings used for data encoding. However, each of these tag types consisted of a single circle pattern used for pose estimation rather than a set of dots used for point correspondences.

In 2011, Bergamasco et al. [7] presented a tag design using a large number of circular dots spaced around a circular ring that also used the work of Chen to simplify the detection of the ring. In 2013, Bergamasco et al. [8] also presented PiTags that used the invariant properties of projective geometry to detect tags made of ellipses spaced in a square pattern. Though the appearance of our tag border most closely resembles that of PiTags, we do not use their presented detection methods. The detection methods they present restict the tag to have exactly 12 points and can be slow and difficult to detect at far range. Our method allows us to leverage the detection qualities of any desired state-of-the-art tag, and allow us to vary the number of points used in the estimation process, thus trading off between accuracy and speed.

The experimental tests and comparison methods presented in these papers evaluate localization accuracy under: occlusion, noise, Gaussian blur, viewing angle, distance, and illumination gradient using simulated images [3, 5, 7]. The detection methods and pose estimation algorithms presented in these papers are also often validated using a small number of indoor images. However, to our knowledge, none of the experimental tests presented in the literature include testing or evaluation with real-imagery exposed to high variance outdoor lighting.

Once a tag is detected, the pose of the tag in the camera frame is estimated. A common method of doing so is by attempting to minimize the reprojection error of 3D-to-2D point correspondences returned by the tag detection algorithm. This is often referred to as the Perspective-n-Point problem in computer vision and photogrammetry. A variety of methods have been proposed to solve this problem including those outlined in [3], [9], and the popular Levenberg-Marquardt algorithm [10].

To use the pose measurement in a filtering framework such as a Kalman Filter, it is also important to accurately estimate the uncertainty of these pose measurements. This uncertainty is dependent on tag pose as well as the uncertainty of the estimated 2D pixel point detections. It is common to model each of these estimated values as jointly Gaussian random variables and characterize uncertainty by estimating the covariance matrix of these distributions. In his seminal paper, Haralick [11] suggests the propagation of pixel noise as a method of estimating parameter covariance, we follow that approach here.

III. Background Information

Here we present some necessary background information for understanding our methodology and notation.

A. Coordinate Frames and Relative-Pose Operations

The camera and tag each have a predefined coordinate frame. We denote these frames by c and t, respectively. We seek to estimate the relative-pose offset of the tag relative to the camera.

For relative-pose offsets, we adopt the methods used in [12, 13]. Specifically, we define the representation of frame j with respect to frame i as

$$\begin{aligned}\boldsymbol{x}_{ij} &= \begin{bmatrix} {}^i\boldsymbol{t}_{ij}^\top & \boldsymbol{\Theta}_{ij}^\top \end{bmatrix}^\top \\ &= \begin{bmatrix} x_{ij} & y_{ij} & z_{ij} & \phi_{ij} & \theta_{ij} & \psi_{ij} \end{bmatrix}^\top,\end{aligned}$$

where the homogeneous transformation matrix from frame j to frame i is

$${}^i_j\boldsymbol{H} = \begin{bmatrix} {}^i_j\boldsymbol{R} & {}^i\boldsymbol{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix}.$$

Here, ${}^i\boldsymbol{t}_{ij}^\top = [x_{ij}, y_{ij}, z_{ij}]^\top$ is the translational offset from the i-th frame to the j-th frame as expressed in the i-th frame and ${}^i_j\boldsymbol{R}$ is the SO3 rotation matrix that brings a point defined in the j-th frame into the i-th frame using the following convention for Euler angles

$${}^i_j\boldsymbol{R} = \text{rotxyz}(\boldsymbol{\Theta}_{ij}) = \text{rotz}(\phi_{ij})^\top \text{roty}(\theta_{ij})^\top \text{rotx}(\theta_{ij})^\top.$$

In addition, we adopt the notation of [14] to denote the relative-pose offset composition operation as

$$\boldsymbol{x}_{ik} = \boldsymbol{x}_{ij} \oplus \boldsymbol{x}_{jk},$$

and the relative-pose offset inverse operation as

$$\boldsymbol{x}_{ji} = \ominus \boldsymbol{x}_{ij}.$$

B. Pinhole Camera Projection Model

The pinhole projection camera model is based on the assumption that rays of light reaching the image frame all travel through a single point. Though a simplification, it is commonly used in the robotics and computer vision community and works quite well for cameras with small apertures and when the scene depth is in focus [15].

In this model, the 3×4 camera projection matrix $\boldsymbol{P}$ maps homogeneous 3D points expressed in an arbitrary world coordinate frame into the 2D image frame. This can be seen as follows

$$\boldsymbol{u'} = \boldsymbol{P}\ {}^{w}\boldsymbol{X'}, \tag{1}$$

where $\boldsymbol{u'}$ is a homogeneous representation of the 2D pixel coordinates, ${}^{w}\boldsymbol{X}$ is a 3D point expressed in the world coordinate frame, and ${}^{w}\boldsymbol{X'}$ is the homogeneous represenation of the point ${}^{w}\boldsymbol{X}$.

The projection matrix $\boldsymbol{P}$ can be further decomposed into an intrinsic parameters matrix $\boldsymbol{K}$ and an extrinsic parameters matrix $\left[\ {}^{c}_{w}\boldsymbol{R} \mid {}^{c}\boldsymbol{t}_{cw}\ \right]$,

$$\boldsymbol{u'} = \boldsymbol{K}\ \left[\ {}^{c}_{w}\boldsymbol{R} \mid {}^{c}\boldsymbol{t}_{cw}\ \right]\ {}^{w}\boldsymbol{X'}. \tag{2}$$

The extrinsic parameters matrix changes the representation of the world point so that is expressed in the camera frame (${}^{c}\boldsymbol{X}$) and then the intrinsic parameters matrix $\boldsymbol{K}$ projects it into the 2D image frame. $\boldsymbol{K}$ is a 5 degree of freedom (DoF) matrix whose parameters can be determined through camera calibration [16]. If we know the position of the 3D world points with respect to the camera frame then we can take the extrinsic parameters matrix to be $\left[\ \boldsymbol{I} \mid \boldsymbol{0}\ \right]$ and only need to multiply by $\boldsymbol{K}$ to determine its pixel coordinates in the image frame.

In addition, note that because the camera imaging process loses the degree of freedom corresponding to depth, the camera matrix $\boldsymbol{P}$ is only defined up to scale and has 11 DoFs. Thus, the pixel coordinates $(x,\ y)$ of the point ${}^{w}\boldsymbol{X'}$ are equal to $(u/s,\ v/s)$, where $\boldsymbol{u'} = [u, v, s]^{\top}$ in accordance with the definition of homogeneous coordinates [15].

IV. Visual Fiducial Pose Estimation

The software pipeline for the proposed system consists of tag detection, pose estimation, uncertainty estimation, and filtering. When an image is received from the camera, we run a detection algorithm to locate the tag within the image if present. Once a detection has been made, we determine the 2D pixel coordinates that correspond to a pre-determined set of 3D points known relative to the tag coordinate frame. Using projective geometry and optimization, we can then estimate the pose of the tag relative to the camera frame of reference by minimizing reprojection error. Once a tag pose estimate has been determined, we estimate the uncertainty of the measurement so that it can be integrated into a probabilistic filtering framework such as a Kalman filter.

A. Estimating Tag Pose

Each state-of-the-art fiducial framework provides a set of tag designs and an algorithm for detecting them in a given image. Most designs include a robust method for uniquely identifying the tag and a method for estimating the pose of the tag with respect to the camera. In our system, we opted to use the updated version of the AprilTag library [17] because of its high-speed detection algorithm and open-source implementation. However, in our experiments we determined that the DLT algorithm provided in AprilTags for tag pose estimation results in a very noisy pose estimate. This is because the DLT algorithm does not restrict the estimated rotation matrix to be orthonormal and uses a polar decomposition to find the closest valid rotation matrix (in terms of the Frobenius matrix norm) [3]. Rather than use the faster but less accurate DLT algorithm, we opted to use an iterative method to simultaneously enforce orthonormality and directly minimize reprojection error. The rest of this section explains how to set up this optimization problem.

The goal of the tag pose estimation problem is to estimate the relative-pose offset of the tag coordinate frame with respect to the camera frame $\boldsymbol{x}_{ct}$.

As explained in (2), if we know the camera calibration matrix $\boldsymbol{K}$, the extrinsic parameter matrix $\left[\ {}^{c}_{t}\boldsymbol{R} \mid {}^{c}\boldsymbol{t}_{ct}\ \right]$ that encodes the pose of the tag relative to the camera, and a point ${}^{t}\boldsymbol{X}^{[i]}$ expressed in the tag frame, we can calculate its pixel coordinates $\boldsymbol{u}^{[i]}$ as shown

$$\boldsymbol{u'}^{[i]} = \begin{bmatrix} u \\ v \\ s \end{bmatrix} = \boldsymbol{K}\ \left[\ {}^{c}_{t}\boldsymbol{R} \mid {}^{c}\boldsymbol{t}_{ct}\ \right]\ {}^{t}\boldsymbol{X'}^{[i]} \tag{3}$$

where $\boldsymbol{u}^{[i]} = [u/s, v/s]^{\top}$.

In tag pose estimation however, ${}^{c}_{t}\boldsymbol{R}$ and ${}^{c}\boldsymbol{t}_{ct}$ are unknown and our goal is to estimate their parameters $\boldsymbol{x}_{ct}$. We determine 2D-3D point correspondences by matching 2D detected image coordinates $\{\boldsymbol{u}^{[i]}\}_{i=1}^{N}$ to pre-determined 3D tag points $\{{}^{t}\boldsymbol{X}^{[i]}\}_{i=1}^{N}$ and then seek to estimate $\boldsymbol{x}_{ct}$ by minimizing the reprojection error

$$\hat{\boldsymbol{x}}_{ct} = \underset{\boldsymbol{x}_{ct}}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| \boldsymbol{u}^{[i]} - \hat{\boldsymbol{u}}^{[i]} \right\|^{2} \tag{4}$$

where $\hat{\boldsymbol{u}}^{[i]} = f({}^{t}\boldsymbol{X}^{[i]}, \boldsymbol{x}_{ct})$ is the function that dehomogonizes the result of the following matrix multiplication $\boldsymbol{K}\ \left[\ {}^{c}_{t}\boldsymbol{R} \mid {}^{c}\boldsymbol{t}_{ct}\ \right]\ {}^{t}\boldsymbol{X'}^{[i]}$ by normalizing the first and second elements by the third.

There are many algorithms available for solving non-linear least squares problems like this one. We opted to use the Levenberg-Marquardt option built into the OpenCV function *solvePnP* [10, 18].

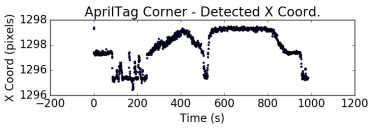
B. Blooming Bias and Proposed Tag Extension

The method presented in §IV-A is much more accurate than the DLT algorithm presented in [3], but is also dependent on good estimates of the 2D-3D point correspondences. Many state-of-the-art tag designs including, ARToolkit, ARTag, and AprilTag [1–3], seek to estimate the

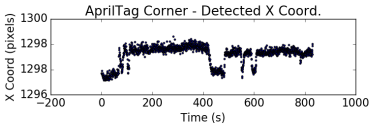
(a) 2000μs Exposure Time

(b) 500μs Exposure Time

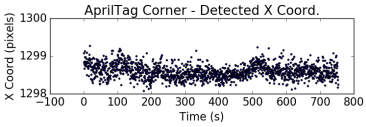
(c) 250μs Exposure Time



(d) 2000μs Detected Coordinate

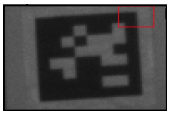


(e) 500μs Detected Coordinate

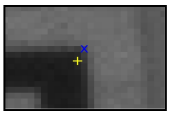


(f) 250μs Detected Coordinate

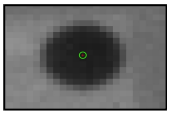
Fig. 4. Here we show that though decreasing exposure time does decrease the effects of blooming, there is still significant biasing even at short exposure times such as 500μs. Plots (d), (e), (f) show detected x pixel coordinates of a static tag corner. Note that an exposure time of 250μs does diminish the biasing effects, but this severe decrease in exposure time increases noise and darkens the image enough that it begins to affect tag detectability.



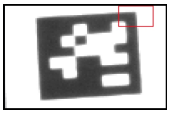
(a) AprilTag - Norm



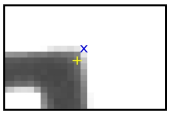
(b) Corner - Norm



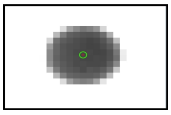
(c) Circle - Norm



(d) AprilTag - Bright



(e) Corner - Bright



(f) Circle - Bright

Fig. 2. This figure highlights the effects of blooming and saturation on tag detection and point estimation. For each pair of images (taken just a few seconds apart), the camera and tag are entirely static, however there is a notable difference in the captured image. The colored marks are given as static reference points in pixel space between the matching images. This change in the image causes biasing of detected corner coordinates.

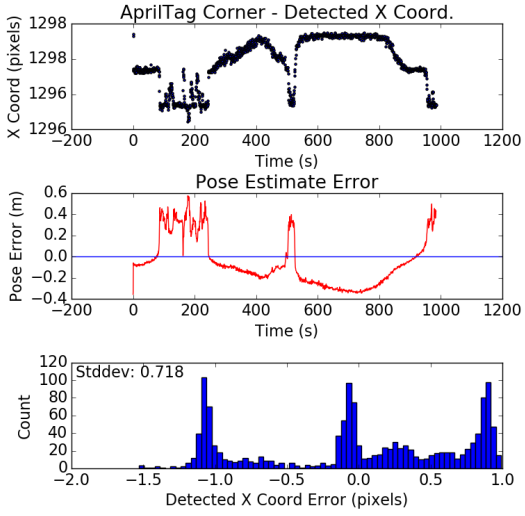


Fig. 3. For a static tag and camera, the detected AprilTag corners vary significantly when the changes in outdoor lighting induce blooming. This bias translates to a significant bias and error in the estimated pose of the tag as can be seen in the middle plot. The histogram plot illustrates that the detection error is certainly non-Gaussian. In this experiment, the lighting of the sun changed at approximately time steps 80, 220, 440, 820, and 970. Error is taken with respect to the mean.

four outer corners of a black square with a white background. However, in naturally varying sunlight, we observed a significant bias in the estimation of these corner coordinates caused by blooming of the CCD. Blooming describes the phenomenon when an overabundance of light causes the white areas of the image to bloom out into the surrounding areas, see Fig. 2. In our tests, blooming caused the detected points to vary by several pixels, which in turn resulted in the estimated pose varying by up to a meter, see Fig. 3. In addition, Fig. 4 shows that this error cannot be solved by varying exposure time alone. As exposure time decreases, the image darkens and the blooming effects are diminished, but not enough to be disregarded. In addition, as the image becomes darker the signal to noise ratio is increased and tag detection becomes more difficult.

The significant bias is caused because the AprilTag algorithm detects corners at the intersection of lines around the outer edge of the square. When the blooming occurs, the estimates of the lines in the image are pushed inward resulting in a significant change in the estimated corner coordinates. Estimating the center of a circle, on the other hand, is more invariant to the effects of blooming because the border of the circle is affected relatively equally, see Fig. 2(c) and (f).

We propose an extension to tags affected by blooming by placing circles around the outer border of the tag as shown in Fig. 1(b). By using the original fiducial detector points to find an initial guess of the tag pose, we can create 2D-3D point correspondences for the ellipse centers. Then using these correspondences we can determine a blooming-robust estimate of the final pose using the more accurate ellipse point correspondences.

If we let $\{\boldsymbol{u}_f^{[i]}\}$ be the set of 2D tag points returned by the fiducial detection algorithm, let $\{{}^t\boldsymbol{X}_f^{[i]}\}$ be the corresponding set of 3D tag points, and let $\hat{\boldsymbol{x}}_{cf}$ be the pose of the tag with respect to the camera frame as estimated from the fiducial tag points, then we can estimate $\hat{\boldsymbol{x}}_{cf}$ by evaluating (4) over $\{\boldsymbol{u}_f^{[i]}\}$ and $\{{}^t\boldsymbol{X}_f^{[i]}\}$.

Then letting $\{{}^t\boldsymbol{X}_e^{[i]}\}$ be the set of 3D ellipse center points in the tag frame, we can use $\hat{\boldsymbol{x}}_{cf}$ to reproject these points back into the image frame and obtain a hypothesis set of pixel locations for the ellipse centers $\{\hat{\boldsymbol{u}}_{er}^{[i]}\}$

$$\hat{\boldsymbol{u}'}_{er}^{[i]} = \boldsymbol{K} \left[\begin{array}{c|c} {}^c_f\boldsymbol{R} & {}^c\boldsymbol{t}_{cf} \end{array} \right] {}^t\boldsymbol{X'}_e^{[i]}. \tag{5}$$

Now with an initial guess of the ellipse centers, we can binarize the important portion of the image, extract possible ellipse contours, and fit ellipses to the valid contours to obtain a high accuracy estimate of their centers $\{\boldsymbol{u}_e^{[i]}\}$. We can then form 2D-3D ellipse point correspondences by performing data association to match the detected ellipse centers $\{\boldsymbol{u}_e^{[i]}\}$ to their respective 3D tag points using their reprojected 2D coordinates $\{\hat{\boldsymbol{u}}_{er}^{[i]}\}$. In our application a nearest neighbor approach worked well, but it would also be possible to estimate these matches jointly if necessary.

Finally, if data association succeeds, we can estimate the final pose directly from the matched ellipse correspondences by evaluating (4) over the detected ellipse centers $\{\boldsymbol{u}_e^{[i]}\}$ and their associated 3D points in the tag frame $\{{}^t\boldsymbol{X}_e^{[i]}\}$.

To save time we can use the estimate $\{\boldsymbol{u}_e^{[i]}\}$ from the last iteration as an estimate of $\{\hat{\boldsymbol{u}}_e^{[i]}\}$ depending on the camera frame rate.

The final algorithm is shown in Algorithm 1, where the functions `project_ell_pnts` and `project_fid_pnts` are wrapper functions that estimate the pose of the tag given the passed in points and then call `project_points` to project either the ellipse points or the fiducial points, respectively, back into the image frame. The `project_points` function implements (3) where ${}^t\boldsymbol{X'}^{[i]}$ is parameterized by $\boldsymbol{x}_{tp}^{[i]}$ with the rotation parameters equal to 0. The `solve_pose` function implements equation (4).

The number and size of the circles are parameters that should be chosen dependent on the application. The tradeoffs in selecting the number of dots will be further explored in §VI-B. The size of the dots should be chosen dependent on the expected range to the target and the camera resolution. The dots should be large enough that they can be accurately and consistently detected in outdoor lighting, but dots that are too large can introduce biasing problems when trying to estimate the center of the ellipse at obscure angles [19].

Algorithm 1 Relative-Pose Tag Estimation Algorithm

Require: $valid\_det_{t-1}$ = False

while True **do**
  wait for image I_t
  I_t = `undistort_image`(I_t)
  if $valid\_det_{t-1}$ is False **then**
    $\{\boldsymbol{u}_f^{[i]}\}_t$ = `fiducial_detector`(I_t)
    ROI = `calculate_ROI`($\{\boldsymbol{u}_f^{[i]}\}$, I_t)
    $\{\hat{\boldsymbol{u}}_{er}^{[i]}\}$ = `project_ell_pnts`($\{< \boldsymbol{u}_f,\ {}^t\boldsymbol{X}_f >^{[i]}\}_t$)
  else
    ROI = `calculate_ROI`($\{\boldsymbol{u}_f^{[i]}\}_{t-1}$, I_t)
    $\{\hat{\boldsymbol{u}}_{er}^{[i]}\}$ = `prop_ell_pnts`($\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_{t-1}$)
  end if
  $contours$ = `find_contours`(ROI)
  $\{\boldsymbol{u}_e^{[i]}\}$ = `fit_ellipses`($contours$)
  $\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_t$ = `data_assoc`($\{\boldsymbol{u}_e^{[i]}\}$, $\{\hat{\boldsymbol{u}}_{er}^{[i]}\}$)
  if $\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_t$ is valid **then**
    $\boldsymbol{x}_{ct}$ = `solve_pose`($\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_t$)
    $\boldsymbol{\Sigma}_{ct}$ = `est_uncertainty`($\boldsymbol{x}_{ct}$, $\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_t$)
    `update_filter`($\boldsymbol{x}_{ct}$, $\boldsymbol{\Sigma}_{ct}$)
    $valid\_det_t$ = True
    $\{\boldsymbol{u}_f^{[i]}\}_t$ = `project_fid_pnts`($\{< \boldsymbol{u}_e,\ {}^t\boldsymbol{X}_e >^{[i]}\}_t$)
  else
    $valid\_det_t$ = False
  end if
  t = t + 1
end while

Function: $\{Points\}$ = `project_points`($\boldsymbol{x}_{ct}$, $\{\boldsymbol{x}_{tp}^{[i]}\}_i$)

for ($\boldsymbol{x}_{tp}^{[i]}$ in $\{\boldsymbol{x}_{tp}^{[i]}\}_i$) **do**
  $\boldsymbol{x}_{cp} = \boldsymbol{x}_{ct} \oplus \boldsymbol{x}_{tp}^{[i]}$
  ${}^c\boldsymbol{X} = [x_{cp}, y_{cp}, z_{cp}]^\top$
  $\boldsymbol{u'} = \boldsymbol{K}\ {}^c\boldsymbol{X}$
  $\{Points\} \leftarrow [u/s, v/s]^\top$
end for

C. Measurement Uncertainty Estimation

Once we have an accurate tag pose estimate, we use an extended Kalman filter (EKF) to filter the measurements. However, this requires an understanding of the pose estimate uncertainty. The method we propose for uncertainty characterization is based on the propagation of Gaussian pixel noise through the non-linear pose estimation process. Pixel noise refers to small fluctuations in the estimated location of 2D points in the image frame. If we can find an estimate of the uncertainty in these 2D point detections, we can propagate that uncertainty through the functions used to estimate pose and determine an estimate of the pose measurement uncertainty. In our initial experiments we assumed this pixel noise to be of constant variance and that the two coordinates were independent. However, if desired these values can be estimated on a per measurement basis using the methods explained by Ji and Haralick [20]. We present two methods for propagating this uncertainty.

The first is commonly referred to as backward propagation and gives a first-order approximation of the estimated covariance as explained in [15]. In this case, given an estimate of the covariance of a set of pixel points, $\boldsymbol{\Sigma}_{pix}$, and an estimated 6-DOF pose $\boldsymbol{x}_{ct}$, we can estimate the covariance of the detected pose $\boldsymbol{\Sigma}_{ct}$ according to

$$\boldsymbol{\Sigma}_{ct} = (\boldsymbol{G}^\top \boldsymbol{\Sigma}_{pix}^{-1} \boldsymbol{G})^{-1},$$

where $\boldsymbol{G}$ is the Jacobian of the function `project_points` evaluated on the ellipse points with respect to $\boldsymbol{x}_{ct}$.

The second method is through use of the unscented transform developed by Julier [21]. Though the first-order approximation is fast and sometimes accurate enough, the pose estimation function is highly non-linear and can be better approximated by generating deterministic sigma points

from the detected 2D points in pixel space and estimating pose based on each of them individually. The estimated pose covariance is then determined by taking a weighted average according to the following equation

$$\boldsymbol{\Sigma}_{ct} = \sum_{i=0}^{2n} \omega_c^{[i]} (\boldsymbol{x}_{ct}^{[i]} - \boldsymbol{\mu}')(\boldsymbol{x}_{ct}^{[i]} - \boldsymbol{\mu}')^\top,$$

where $\omega_c^{[i]}$ is a pre-determined weight, $\boldsymbol{x}_{ct}^{[i]}$ is the estimated pose given the ith sigma point, and $\boldsymbol{\mu}'$ is a weighted average of the $\boldsymbol{x}_{ct}^{[i]}$. More information can be found in [21–23].

V. Ship Pose Estimation and Filtering

In order to integrate our tag detection system into a relative ship pose estimation system, we rigidly mounted a camera to one ship and tag printed on aluminum signing to the other ship. We defined coordinate frames for the two ships at the approximate centers of gravity specified by A for the camera ship and B for the tag ship, respectively.

The goal of the system is to estimate the relative-pose of the two ships $\boldsymbol{x}_{ab}$ and this denotes our filter state. We assume the pose of the camera and tag relative to their respective ships $\boldsymbol{x}_{ac}$ and $\boldsymbol{x}_{bt}$ are known. Our system directly estimates $\boldsymbol{x}_{ct}$ and these measurements need to be translated to the ship pose frames to estimate $\boldsymbol{x}_{ab}$, see Fig. 1(b).

Thus our observation model is:

$$\hat{\boldsymbol{x}}_{ct} = (\ominus \boldsymbol{x}_{ac}) \oplus \boldsymbol{x}_{ab} \oplus \boldsymbol{x}_{bt} + \boldsymbol{\omega}$$

where $\boldsymbol{\omega} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{ct})$. See [14] for more information on covariance propagation in this context.

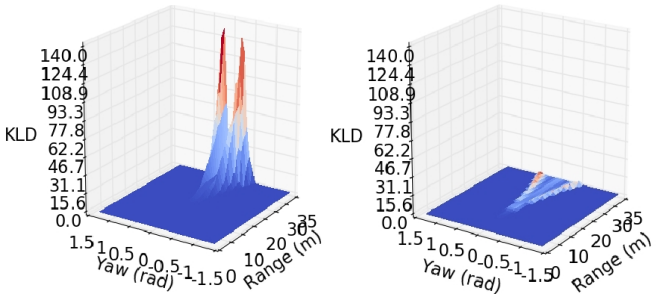
VI. Results

For our testing and system, we used a Prosilica GT2450 GigE mono-chromatic camera with a resolution of 2448 x 2050 pixels and a 12mm fixed focal length lens. In initial testing (§VI-A) we used a Lenovo W540 laptop with an Intel Quad-Core i7-4900MQ 2.80GHz CPU processor and an NVIDIA Quadro K2100M video card used for image rectification. In the final system (§VI-C) we used a Dell Latitude 14 Rugged Extreme Laptop with an Intel Quad-Core i7-4650U 1.70GHz processor and GeForce GT 720M again used for image rectification.

A. Proposed Tag Extension Evaluation

We tested our proposed tag extension on outdoor imagery in dynamic lighting by taking a sequence of images of two static tags of comparable size, one with our extension and one without it. For each image, we then used Levenberg-Marquardt to minimize reprojection error of the AprilTag corners or our ellipse centers respectively and looked at the estimate pose error over time. In this experiment, the tags are scaled so that the distance from one corner point to the other along an edge was 0.70 meters and we took sets of images at 18 meter and 30 meter ranges at a rate of 4 Hz.

Fig. 5(a) and Fig. 5(b) show that in the standard AprilTag case, the estimated pose of the static tag varies on the order of a meter, while the estimated pose of the extended tag only varies on the order of about 2 centimeters. Similarly, at 30 meters, the pose went from varying on the order of 5 meters without our extension to 5 centimeters with it.



(a) KLD of FO from MC (b) KLD of UT from MC

Fig. 6. Here we show the Kullback Leibler Divergance (KLD) of the first-order (FO) and unscented transform (UT) methods from a Monte-Carlo (MC) simulated distribution. The divergance is shown at 1600 pose locations varying over both range from the camera to the tag and the angle of incidence with the camera principle axis. Note that some of these poses are outside the camera field of view. The divergance of the unscented transform is significantly lower than that of the first order approximation. These results are for an 8 point tag.

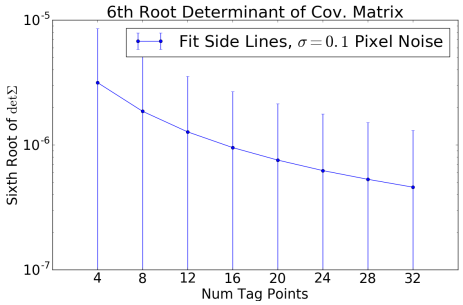


Fig. 7. Here we show a summary of overall pose uncertainty versus the number of tag points of the Monte-Carlo simulation explained in §VI-B. The sixth root determinant of a pose covariance matrix is a measurement of overall pose uncertainty. The mean sixth root determinant over poses is shown along with 3-sigma error bars.

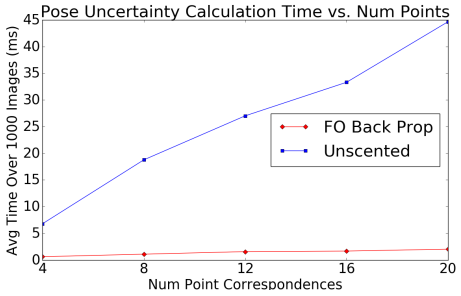
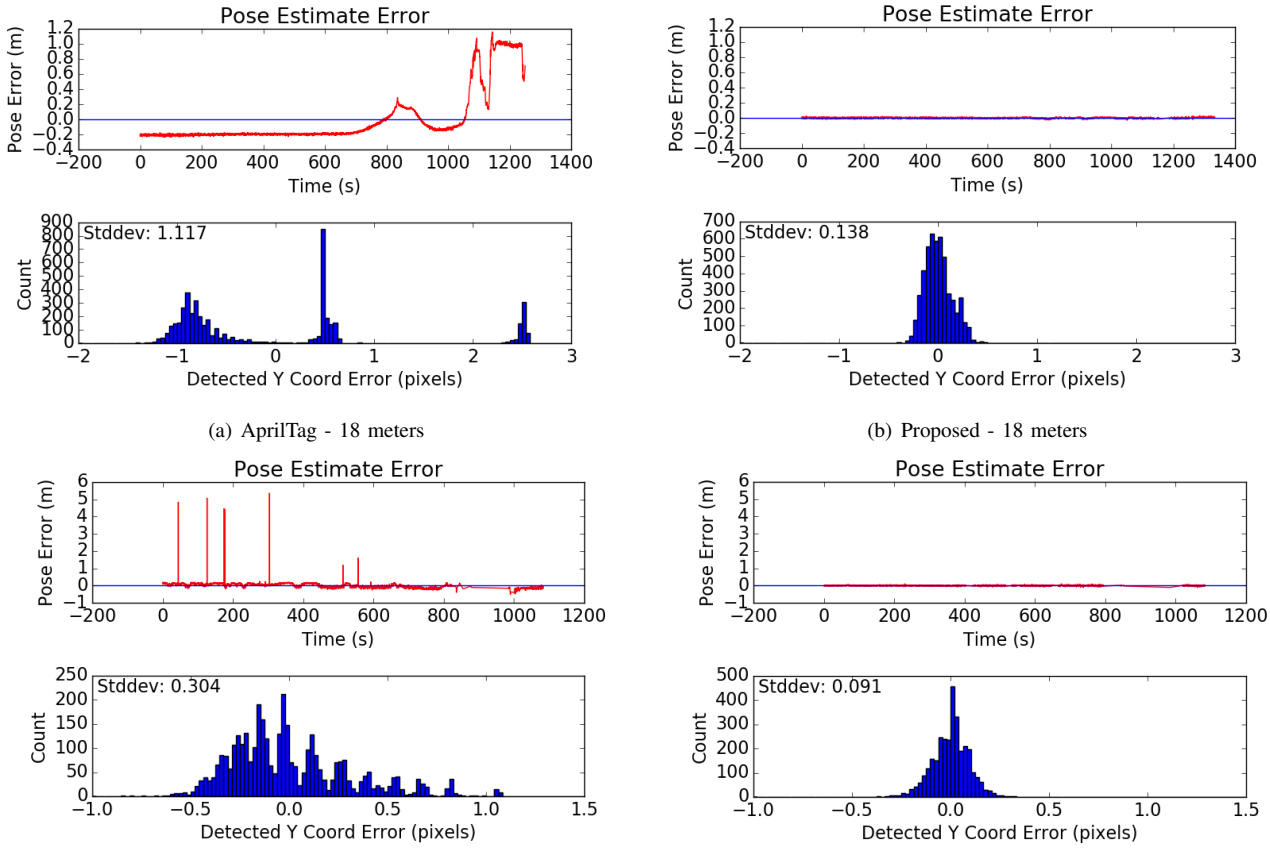


Fig. 8. Here we show the increase in average processing time of the two methods versus the number of tag points used. The increase in time is significantly more pronounced for the unscented transfrom because the pose is estimated for each sigma point set of pixel coordinates.

B. Uncertainty and Processing Time Analysis

We also performed some analysis looking at the trade-offs between the first-order (FO) and unscented transform (UT) uncertainty characterization methods.



(a) AprilTag - 18 meters

(b) Proposed - 18 meters

(c) AprilTag - 30 meters

(d) Proposed - 30 meters

Fig. 5. Here we show the estimated pose error for a generic AprilTag compared to our proposed extension at range. A generic AprilTag and one with the proposed extension were placed side by side in a static position under dynamic outdoor lighting and a sequence of images were collected. The top plot shows pose estimate error relative to the mean and the second plot shows the distribution of pixel coordinate error from the mean. As can be seen in (a) and (b), at 18 meters range and under the same lighting conditions, the estimated pose goes from varying on the order of one meter to varying on the order of +/- 2 centimeters. At a farther range of 30 meters, the pose goes from varying on the order of +/- 5 meters to +/- 5 centimeters, see (c) and (d).

The unscented transform better approximates the non-linear pose estimation function and thus results in a more accurate estimate of the measured pose uncertainty. In order to investigate this trade off, we performed a Monte-Carlo simulation over a set of 1600 poses with the distance from the camera to the tag ranging from 0 to 35 meters and orientation of the tag varying from -80 to 80 degrees off axis. For each of these poses, we projected the tag points into the image frame and then estimated the covariance at each pose using both the first-order and unscented transform methods. For comparison, we added Gaussian noise to produce 1000 sets of sample pixel coordinates per pose and then estimated pose based on each of those individual sets of pixel coordinates and took the sample mean and covariance of each set of 1000 samples. We then evaluated the Kullback Leibler Divergance (KLD) of the two estimated distributions to the Monte-Carlo distribution for each pose. The KLD is a measure of how well one distribution approximates another. Fig. 6 shows how well the first-order and unscented transform methods approximate the Monte-Carlo distribution versus range and orientation.

In addition, increasing the number of points used to estimate pose decreases the measurement uncertainty while increasing the processing time. This influence on timing is particularly visible in the unscented transform because it performs the entire iterative pose estimation process on each sigma point for a total of $2N + 1$ times, where N is the number of points being used. To investigate this trade-off, we performed the previously described Monte-Carlo simulation for tags with an increasing number of tag points and took the sixth root determinant of the covariance matrix for each pose. The determinant can be interpreted as the volume of the six dimensional covariance ellipsoid and is often used as a measure of pose uncertainty. Fig. 7 shows a summary of this pose uncertainty versus the number of tag points. Fig. 8 shows the increase in processing time with the number of points.

C. Preliminary at Sea Relative Ship Motion Results

Finally, we provide preliminary results showing our system's estimate of the relative ship pose of the USNS Bob Hope and the USNS John Glenn collected during our first skin-to-skin field test in November 2015. Fig. 1(a) shows the USNS Bob Hope and a sister ship to the John Glenn in a similar configuration.

In this experiment a camera was rigidly mounted to the hull of the USNS John Glenn and multiple 20 point tags

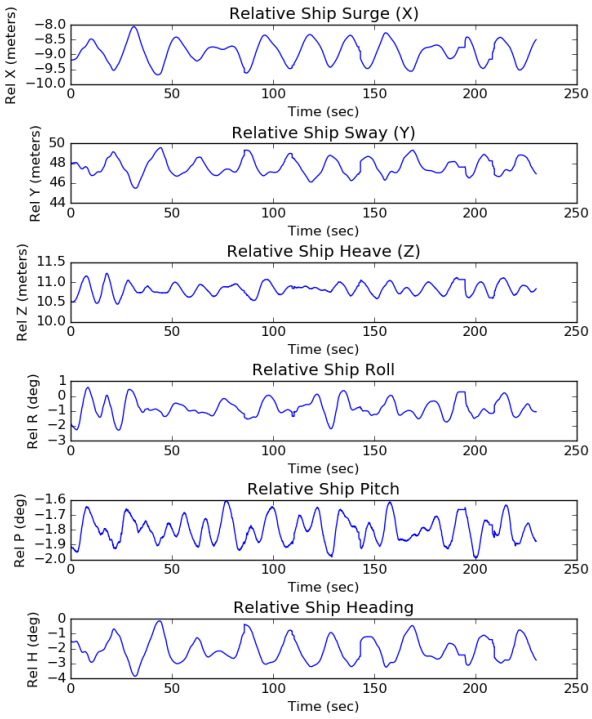


Fig. 9. Here we show the preliminary filtered results of our ship motion system field tests aboard the USNS John Glenn and USNS Bob Hope. These poses were estimated using the four outer ellipse points for speed reasons, though accuracy increases with the number of points.

were mounted to the side of the USNS Bob Hope. The tags were scaled so that the distance between adjacent corner points was 0.93218 meters square and we used a constant pixel variance of 0.1 and assumed independence of x and y for this experiment. Fig. 9 shows the estimated relative pose of the two ships using the unscented transform to estimate measurement uncertainty.

VII. Conclusion

In this paper, we presented a new visual fiducial extension that enables robust pose estimation in varying outdoor lighting. We also presented two methods for characterizing tag pose measurement uncertainty through the propagation of pixel noise. We then presented experimental results showing a dramatic decrease in pose error using our method and provided a discussion of trade-offs between the two uncertainty characterization methods. Finally, we provided preliminary results from skin-to-skin field tests.

In future work, the calibration problem of determining the pose of the tag and camera relative to their respective ship coordinate frames is a difficult and important one. This could possibly be solved with a large number of tags and cameras set up for a one-time intensive calibration proceedure similar to simultaneous localization and mapping. In addition, a good process model for ship motion would significantly improve results.

References

[1] H. Kato and M. Billinghurst, "Marker tracking and HMD calibration for a video-based augmented reality conferencing system," in *Proc. IEEE/ACM Int. Work. Aug. Reality*, San Francisco, California, USA, October 1999, pp. 85–94.

[2] M. Fiala, "ARTag, a fiducial marker system using digital techniques," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 2, San Diego, California, USA, June 2005, pp. 590–596.

[3] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proc. IEEE Int. Conf. Robot. and Automation*, Shanghai, China, May 2011, pp. 3400–3407.

[4] Q. Chen, H. Wu, and T. Wada, "Camera calibration with two arbitrary coplanar circles," in *Proc. European Conf. Comput. Vis.* Prague, Czech Republic: Springer, May 2004, pp. 521–532.

[5] A. C. Rice, A. R. Beresford, and R. K. Harle, "Cantag: an open source software toolkit for designing and deploying marker-based vision systems," in *Proc. IEEE Int. Conf. Perv. Comp. Comm.*, Pisa, Italy, March 2006.

[6] A. Pagani, J. Koehler, and D. Stricker, "Circular markers for camera pose estimation," in *Proc. Int. Work. Image Anal. Mult. Int. Serv.*, Delft, The Netherlands, April 2011.

[7] F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello, "RUNE-Tag: A high accuracy fiducial marker with strong occlusion resilience," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Colorado Springs, Colorado, USA, June 2011, pp. 113–120.

[8] F. Bergamasco, A. Albarelli, and A. Torsello, "Pi-Tag: a fast image-space marker design based on projective invariants," *Mach. Vis. and Applicat.*, vol. 24, no. 6, pp. 1295–1310, 2013.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o (n) solution to the pnp problem," *Int. J. Comput. Vis.*, vol. 81, no. 2, pp. 155–166, 2009.

[10] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, 1963.

[11] R. M. Haralick, "Propagating covariance in computer vision," in *Proc. Int. Conf. Pattern Recog.*, vol. 1, Jerusalem, Israel, October 1994, pp. 493–498.

[12] S. M. Chaves, R. W. Wolcott, and R. M. Eustice, "NEEC research: Toward GPS-denied landing of unmanned aerial vehicles on ships at sea," *Naval Engineers J.*, 2015.

[13] R. M. Eustice, "Large-area visually augmented navigation for autonomous underwater vehicles," Ph.D. dissertation, Department of Ocean Engineering, Massachusetts Institute of Technology / Woods Hole Oceanographic Institution Joint Program, Cambridge, MA, USA, June 2005.

[14] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," *Auton. Robot.*, pp. 167–193, 1990.

[15] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[16] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, Kerkyra, Greece, September 1999, pp. 666–673.

[17] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Daejon, Korea, October 2016, (Under Review).

[18] K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov, "Real-time computer vision with OpenCV," *Comm. ACM*, vol. 55, no. 6, pp. 61–69, 2012.

[19] J. Mallon and P. F. Whelan, "Which pattern? biasing aspects of planar calibration patterns and detection methods," *IAPR Patt. Recog. Letters*, vol. 28, no. 8, pp. 921–930, 2007.

[20] Q. Ji and R. M. Haralick, "Error propagation for computer vision performance characterization," in *Proc. Int. Conf. Image Sci. Syst. Tech.*, vol. 28, Las Vegas, Nevada, USA, June 1999, pp. 429–435.

[21] S. J. Julier, "The scaled unscented transformation," in *Proc. Amer. Control Conf.*, vol. 6, Anchorage, Alaska, USA, May 2002, pp. 4555–4559.

[22] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[23] P. Ozog and R. M. Eustice, "On the importance of modeling camera calibration uncertainty in visual SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, Karlsruhe, Germany, May 2013, pp. 3762–3769.